



Básico de p5.js


hegamos na última unidade com mais JavaScript!

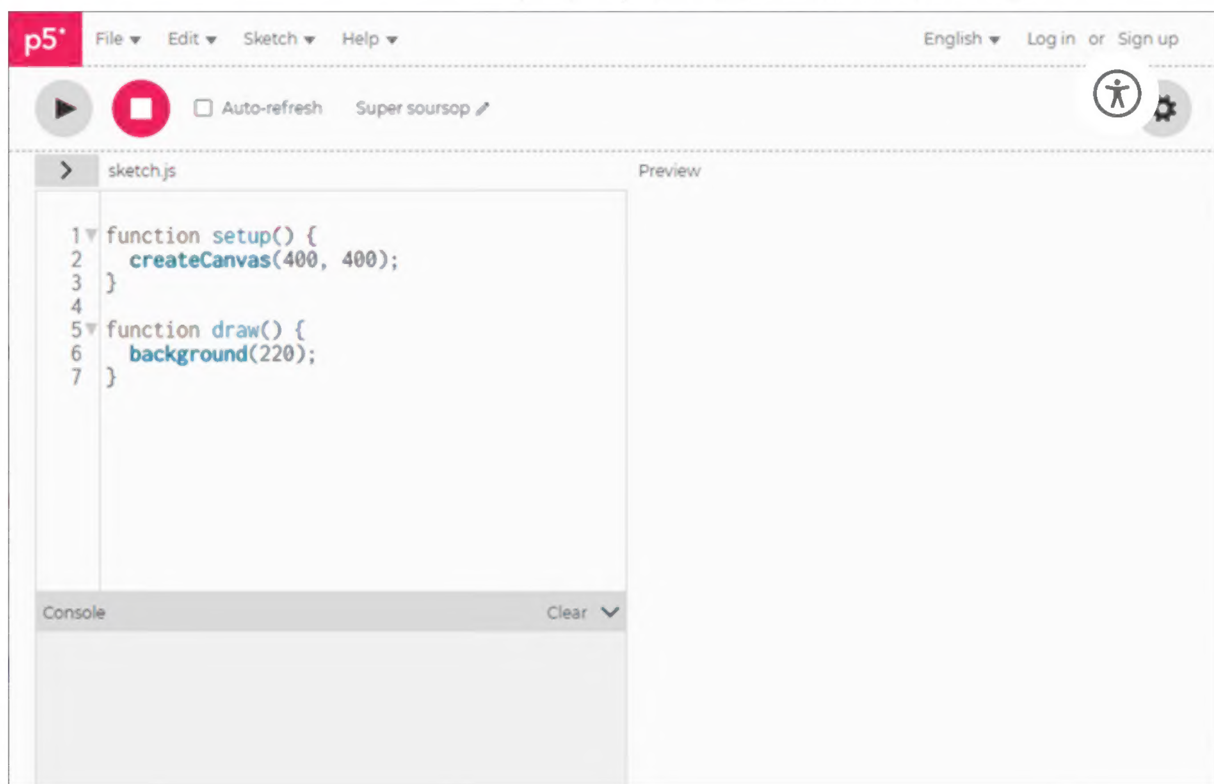
Em nossa última unidade vamos aprender a trabalhar com a biblioteca p5.js. Segundo seu site, a p5.js é uma biblioteca JavaScript para codificação criativa, com foco em tornar a codificação acessível até mesmo para artistas, designers, educadores, iniciantes e qualquer outra pessoa (P5.JS, 2021). Essa biblioteca é muito utilizada para o que chamamos de Computação Criativa, pois permite ampliar criações artísticas aplicadas ao meio digital, incluindo nossos jogos web!

Como é possível notar, o objetivo é ser muito fácil de aprender e usar. Além disso, a p5.js é gratuita e de código aberto. Ela tem uma comunidade muito ativa na internet e sua documentação é ampla, com muita participação dessa comunidade.

Usando a metáfora de *sketch* (esboço em inglês), a biblioteca possui um conjunto completo de funcionalidades de desenho. Ela permite pensar em toda a página do navegador como um *sketch*, incluindo os objetos HTML5 para texto, vídeo, webcam e som.

Funcionamento Básico

A maneira mais fácil de começar é usando o editor p5.js disponível online. Com a ajuda do seu navegador de internet acesse <https://editor.p5js.org>. Você irá encontrar uma janela como a figura abaixo.



Nessa tela você verá um menu, uma barra com os botões de play e stop, à esquerda temos o editor de códigos juntamente com o console e à direita temos a área de prévia, onde veremos o resultado do que estivermos programando no editor.

Se quisermos trabalhar com nosso próprio editor de código e usar a p5.js em nossos projetos, podemos anexar o link para a biblioteca hospedada online através de um CDN. Nesse caso, vamos inserir a seguinte linha de código no documento HTML:

```
<script src="https://cdn.jsdelivr.net/npm/p5@1.4.0/lib/p5.js"></script>
```

Visão Geral do p5.js

Um projeto básico em JavaScript com a biblioteca p5.js possui duas funções:

Ou seja, um projeto básico em JavaScript com a biblioteca p5.js começa com essas duas funções e tem a seguinte estrutura:

- A função *setup()* é executada uma única vez. Normalmente a usaremos para inicialização de variáveis, objetos ou para executar fun , de inicialização;

- A função *draw()* é executada continuamente. A usaremos para realizar uma determinada ação repetidamente e para animações.

```
function setup() {
   // Blocos de código para configurações iniciais do projeto
}
function draw() {
   // Blocos de código para desenhar
}
```

Se abrirmos o editor online da p5.js, vamos nos deparar com o seguinte *sketch*:

```
function setup() {
  createCanvas(400, 400);
}
function draw() {
  background(220);
}
```

Dentro de *setup()*, a função *createCanvas()* cria uma área de desenho no qual a saída é desenhada ou exibida. O código acima cria uma tela de largura e altura de 400 pixels. Já dentro da função *draw()*, tem-se a instrução *background()* que irá preencher a cor de fundo do canvas. Nesse

caso, o valor de 220 no argumento da função refere-se a um tom da cor cinza.



Ao clicarmos no botão de play do editor, aparecerá um quadrado cinza na área de prévia. Vamos agora inserir uma forma básica e uma interação com o mouse.

Interagindo com Elementos Gráficos

Para realizarmos um exemplo neste resumo, vamos fazer um círculo que segue o ponteiro do mouse. Primeiramente, vamos inserir dentro do canvas um círculo na cor verde. Usaremos para isso as funções *fill()* e *ellipse()*.

A função *fill()* define a cor usada para preencher as formas. Basicamente sua sintaxe é dada por:

```
fill(red, green, blue, [alpha]);
```

Onde *red*, *green* e *blue* são valores numéricos relativo ao intervalo de cores RGB e *alpha* é um valor numérico para a transparência do objeto. O Valor de *alpha* é opcional e pode ser omitido se não desejamos transparência.

Para desenharmos um círculo a sintaxe segue o seguinte:

```
ellipse(x, y, width, height);
```

Aqui x e y são as coordenadas do centro da elipse, *width* é a largura e *height* a sua altura.

Então, alterando nosso *sketch* inicial para desenhar o círculo verde com tamanho de 50 pixels (altura e largura) nas coordenadas x = 100 e y = 100, teremos o seguinte trecho de código:

```
function setup() {
  createCanvas(400, 400);
}
function draw() {
  background(220);
  // Define a cor de preenchimento para um tom de verde
  fill(20, 250, 8);
  // Desenha um círculo
  ellipse(100, 100, 50, 50);
}
```

Agora que sabemos desenhar o círculo, vamos fazer ele seguir o ponteiro do mouse. A biblioteca p5.js disponibiliza algumas variáveis de sistema para sabermos a posição do mouse ou para detectarmos cliques, usaremos nesse exemplo as seguintes:

- **mouseX**: contém a posição horizontal atual do mouse, em relação à origem (0, 0) da tela.
- **mouseY**: contém a posição vertical atual do mouse, também em relação à origem (0, 0).
- **mouseIsPressed**: trata-se de uma variável booleana e seu valor é verdadeiro se o mouse for pressionado ou falso caso contrário.

Sabendo disso, vamos alterar nosso *sketch* de exemplo para que o círculo seja desenhado considerando as posições do ponteiro do mo das variáveis *mouseX* e *mouseY.* Além disso, vamos acrescentar uma instrução de decisão para verificar se houve clique do botão do mouse e, caso isso aconteça, a cor de preenchimento do círculo será avermelho.

Vejamos como fica todo o programa:

```
function setup() {
  createCanvas(400, 400);
}
function draw() {
  background(220);
  // Verificar se o mouse foi pressionado
  if (mouseIsPressed) {
    // Se for verdadeiro, ajustar a cor de preenchimento para vermelho
    fill(237, 34, 93);
  } else {
    // Se for falso, a cor de preenchimento será verde
    fill(20, 250, 8);
  }
  ellipse(mouseX, mouseY, 50, 50);
}
```

Finalizamos então os resumos da nossa disciplina de Criação, Navegação e Implementação de Jogos. Nesta última unidade aprendemos sobre uma importante ferramenta para a computação criativa e para ampliar seus conhecimentos aplicados ao desenvolvimento de jogos para a web.

A aplicação dos conhecimentos aprendidos até aqui está agora nas suas mãos. Você está munido de ferramentas para decidir que caminhos seguir em seus projetos. Desejo tudo de bom nessa sua jornada.

Nos encontramos por aí!



Atividade Extra

O canal no YouTube "The Coding Train" tem uma *playlist* completa sobre programação em JavaScript com o p5.js. A *playlist* é um curso online englobando os fundamentos da programação de computadores (variáveis, condicionais, iteração, funções e objetos) usando JavaScript. Em particular, aproveita o ambiente de computação criativa do p5.js. Assista para praticar além do p5.js, boas práticas de programação. Não esqueça de ativar as legendas em português, se preferir!

- **Code! Programming with p5.js**, disponível em: <https://www.youtube.com/playlist?list=PLRqwX-V7Uu6Zy51Q-x9tMWIv9cueOFTFA>.

Referência Bibliográfica

P5.JS - Home. **P5.js.** Disponível em: <https://p5js.org>.

Ir para exercício